NARVA COLLEGE OF THE UNIVERSITY OF TARTU
FACULTY OF SOCIAL SCIENCES
INFORMATION TECHNOLOGY SYSTEMS DEVELOPMENT

Veronika Lahi

PERSONALIZED RECOMMENDATION SYSTEM FOR COUNSELING PROFESSIONALS IN HIGHER EDUCATION

Diploma thesis

Supervisor: Dr. Chen-Wan Lin

NARVA 2021

Olen koostanud töö iseseisvalt. Kõik töö koostamisel kasutatud teiste autorite tööd, põhimõttelised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

..... Töö autori allkiri ja kuupäev

Mina, Veronika Lahi

(sünnikuupäev: 25.06.1995)

annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

PERSONALIZED RECOMMENDATION SYSTEM FOR COUNSELING PROFESSIONALS IN HIGHER EDUCATION,

mille juhendaja on Dr. Chen-Wan Lin,

1.1. reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
1.2. üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace´i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Narvas, **17.05.2021**

PREFACE

Every person attending a university has an opportunity to choose subjects within their study curriculum. Students also must undergo an internship period where they must find a suitable place that fills the requirements set by their university. Due to set criteria, it is particularly important to make a correct decision that is appropriate for their program and will not overwhelm their curriculum. Their practice is the place where students can apply theoretical knowledge in a real-life situation.

The practice provides an opportunity for future employment and development of working relationships. Practice destination also has a set of requirements for students that can complete tasks that are necessary. Therefore, it is important that students choose appropriate subjects, so that the students can reveal their talents and learn something new, as well as improve existing skills.

This diploma work consists of Introduction and Chapter. The Introduction introduces the problem and analyzes the data received from the study group. The Chapter focuses on the system, its structure, as well as the selected technologies. The system's future developmental possibilities are also brought out there.

Appendix 1 stores the files of the developed project.

TERMS AND ABBREVIATIONS

MVP - Minimum Viable Product

UT - University of Tartu

UI - User Interface

MVC - Model-View-Controller

DBMS - Database Management System

SQL - Structured Query Language

INTRODUCTION

This thesis explored the possibility of solving a problem of finding interesting and/or appropriate subjects to study during university study years.

This is an actual topic at the time of this thesis work, because students have difficulties choosing subjects from elective and optional modules while studying at the Narva College of the University of Tartu. An elective module consists of subjects that students can choose from and are related to their curriculum. These subjects help students to narrow down and deepen the knowledge of their study specialty.

Optional module, these are subjects that are unrelated to their study program. Students choose only what they are interested in learning in addition to their curriculum, or subjects that are available but not directly connected to their curriculum.

All universities in Estonia use study information systems, but each university has developed their own individual system. This study focused on the version developed for the University of Tartu. The second version of Tartu University's SIS was finished and released in February 2019.

Students choose their username and after the registration is done by the university, the students choose the subjects that they will study during the upcoming semester. It is necessary to select general subjects, additionally they can choose from the optional subjects, and elective subjects. The student has a list of recommended subjects which have been developed by a study manager to formulate a study plan. This study plan is identical for all students in their study program. Students can register for subjects by finding them in the study information system by name or code.

Also, during the studies, the student will need to finish the "practice" module, the subject is selected in the study information system, but the student must look for practice by themselves. The study information system does not offer practice options.

The limitations of the current SIS used by the University of Tartu include having to manually find subjects of interest, no options for practice, lack of information about the ideal target students.

This research focused on interviewing students and finding the best solution to one of the most popular student problems - the choosing of subjects for elective and optional subjects.

As a result, workable system product will be developed that selects courses based on student preferences.

The Problem

The purpose of this thesis is to make recommendations for further training according to previous professional experience. This work focuses particularly on the course recommendations in higher education in Estonia where students add their own data.

The program will choose suitable courses for the students, based on the needs and hobbies of a particular student. This topic is interesting for the author, as it includes both program development and research aimed at making the program work well and have functions that would distinguish it from similar programs on the market.

In Estonia there are several websites for finding work and practice. These all work on the principle that a person who is looking for a job or a practice, chooses from the proposed options. For example, using a website such as CV.ee or cvkeskus.ee, a choice will be offered either of the entire practices that are available on the website, or all offers of work and practice in total if the person has chosen the direction (for example, IT).

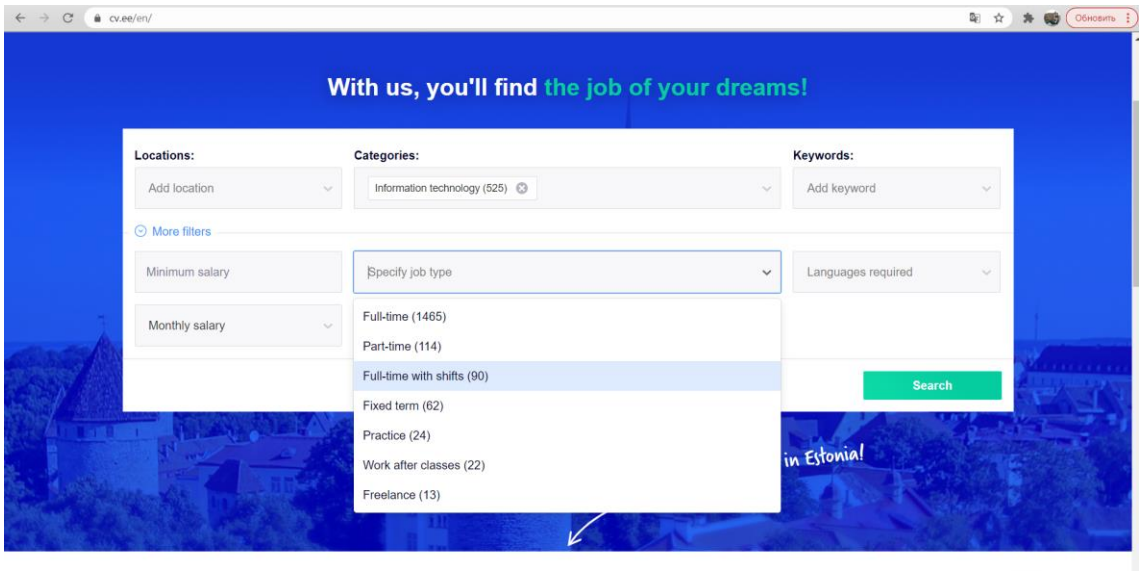

Figure 1. *cv.ee search page.*

From the figure 1 can see that in the IT category there is a division into types of work.

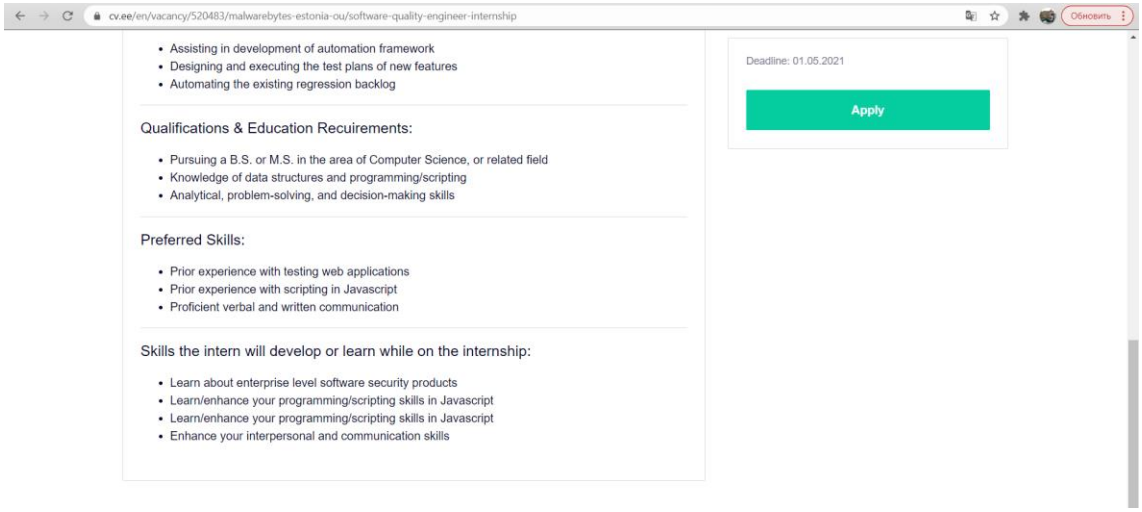

Figure 2. *cv.ee vacancy page.*

As can see in the figure 2, the requirements for a trainee are not clearly written, it is not clear how great the experience should be and what level of knowledge this vacancy requires, also if one's study program is suitable for this practice.

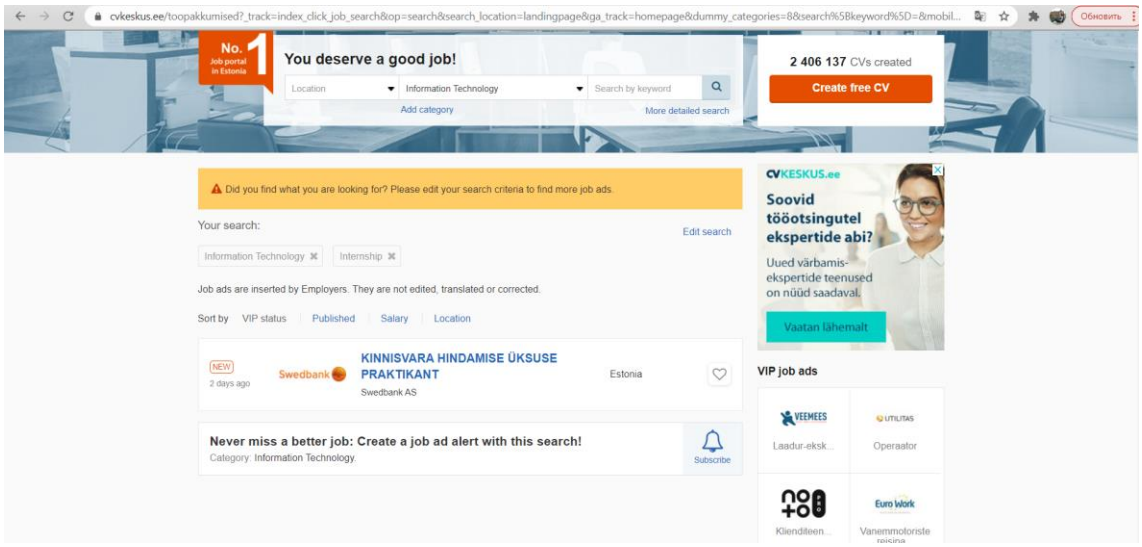

Figure 3. *cvkeskus.ee search page.*

On the next popular website, cvkeskus.ee the result is only one option, compared to a website cv.ee with 24 possibilities. The search for internship vacancies was carried out on the same day 21/04/2021.

Work practice vacancies can often be found only on the company home webpages that offer practice, or at special meetings.

In this system, the program will by itself recommend courses, work, or practice for the user, based on their achievements, successes, and hobbies.

The table below presents a comparison of the existing information system of the University of Tartu and the purpose system.

	Course proposal based on the user data	Possibility for lecturers to create a mock test for students before a course	Practice proposal for students based on the students' data	Opportunity for employees of companies to contact students	See all practice opportunities in one place
TU SIS	No	No	No	No	No
Purpose system	Yes	Yes	Yes	Yes	Yes
cv.ee	-	-	No	Yes	No
cvkeskus.ee	-	-	No	Yes	No

Table 1. *Comparison of systems.*

Also, in the table can see a comparison of the purpose system with the two most popular job search and practice sites in Estonia: cv.ee and cvkeskus.ee

Questionnaire

In the process of working on the graduation project, students of the Narva College of the University of Tartu were interviewed. Total of 25 students from every specialty and different courses. Also, people who graduated from current year or previous year were interviewed, one from each specialty. Due to the coronavirus restrictions, all interviews were conducted over the Internet, using services such as Zoom and Google Meet. In this regard, it should be noted that the students did not give elaborative answers to the questions, which was likely caused by conducting interviews online.

Unfortunately, only a small number of students agreed to participate in the survey. Almost everyone wanted the data to be anonymous. Therefore, in this overview of the data, the author will only use the name of the specialty.

The diagram below shows the specialties and the number of students who participated in the survey. The data is presented as a percentage.

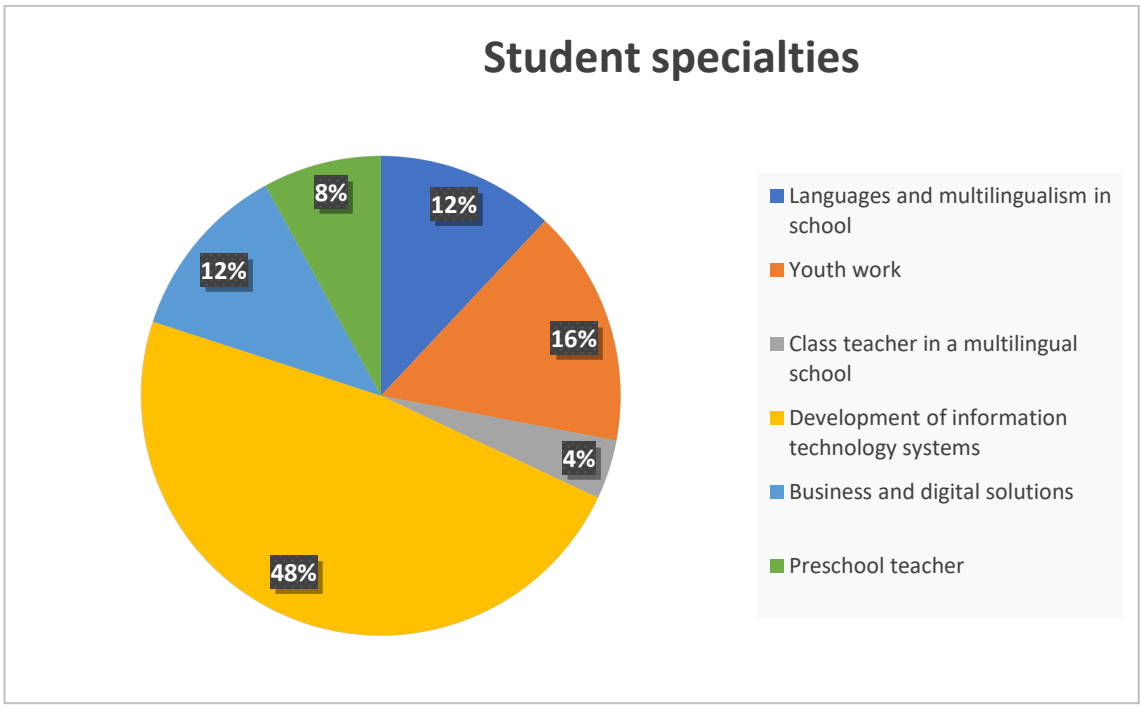

Figure 4. *Specialties of students participating in the survey.*

During research, the author tried to ask at least one person from each specialty and each year of study. But, for example, in the specialty "Primary school teacher", the enrollment of students ended in 2017. Therefore, at that moment, the author was able to ask only two students who were still studying this specialty.

Also, the students of 1-2 years of study were reluctant to share their thoughts. Especially the first-year students, perhaps this is due to the fact that in the last year everyone studied from home, and the students know only people from their group.

Moreover, the humanities students for some reason did not want to share their thoughts, they answered with very general phrases. Students of IT were much more willing to answer the questions and share their thoughts.

The chart below presents data showing how many students from each specialty and each year of study took part in the survey. For example, two graduates, two first-year students, and four students from the second and third years took part in the specialty "Development of Information Technology Systems".

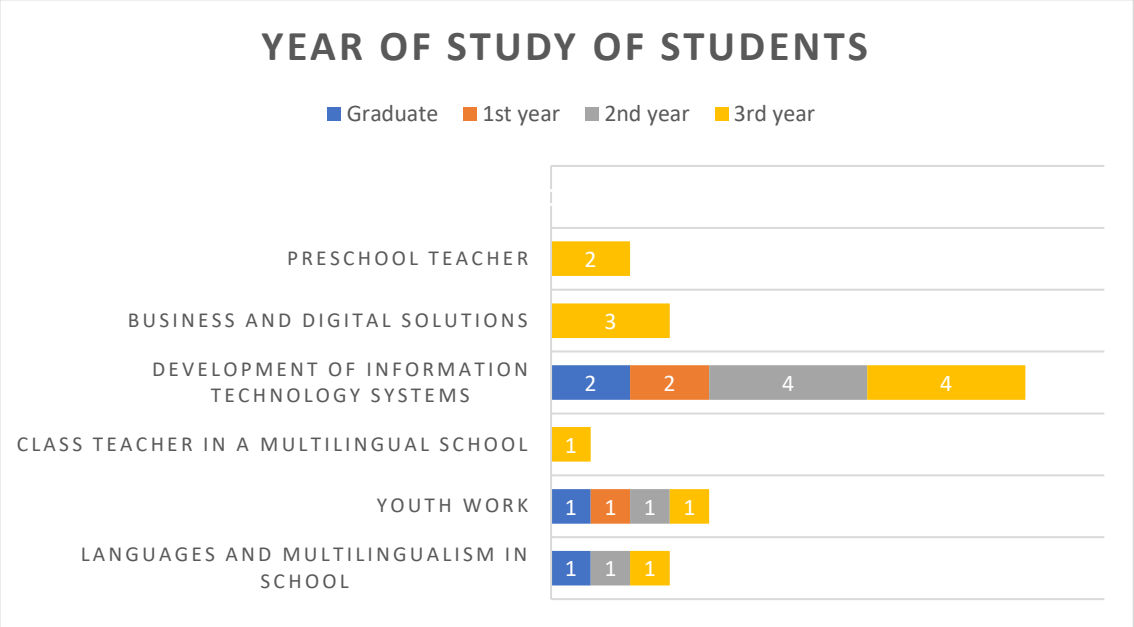

Figure 5. *Year of study of students participating in the survey.*

In the context of this interview, the conversation was about the subjects that students take at the Narva College of the University of Tartu. It is worth noting that the students can choose themselves only the subjects that are in the modules, "Elective subjects" as well as "Optional courses". Most of the subjects in the curriculum are compulsory. Therefore, the conversation did not touch upon the obligatory subjects.

The first four questions were related to the courses.

To the question: Did you remove the subject from the module "Elective subject" or "Optional courses" because it was not interesting for you or did not suit you in terms of knowledge level, only 20 percent of the students answered affirmatively.

The diagram below shows how students answered this question. The diagram contains the students' years of study and their answers, yes or no.

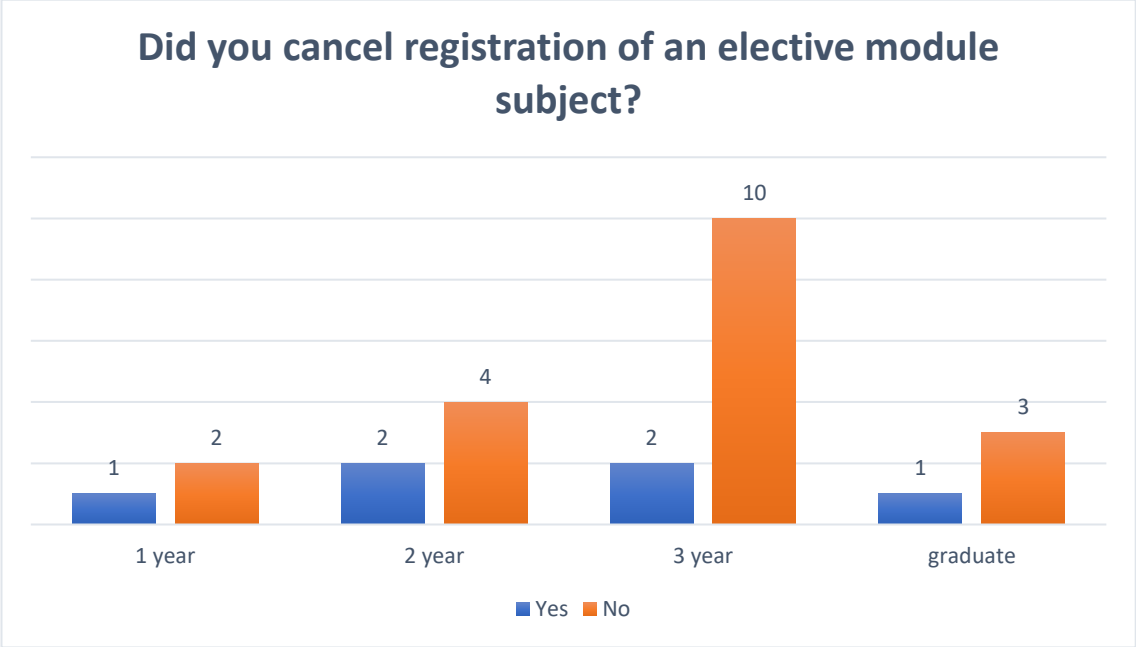

Figure 6. *Respondents' answers to the question.*

At the same time, to the question: "Would you cancel the subject from the module "Elective subject" or "Optional courses" if it did not affect the EAP or the curriculum?" 100% answered in the affirmative.

This data is shown in the diagram below. At the same time, many students from different specialties noted that they made a mistake when choosing a subject but could not cancel it because this would affect their curriculum and results.

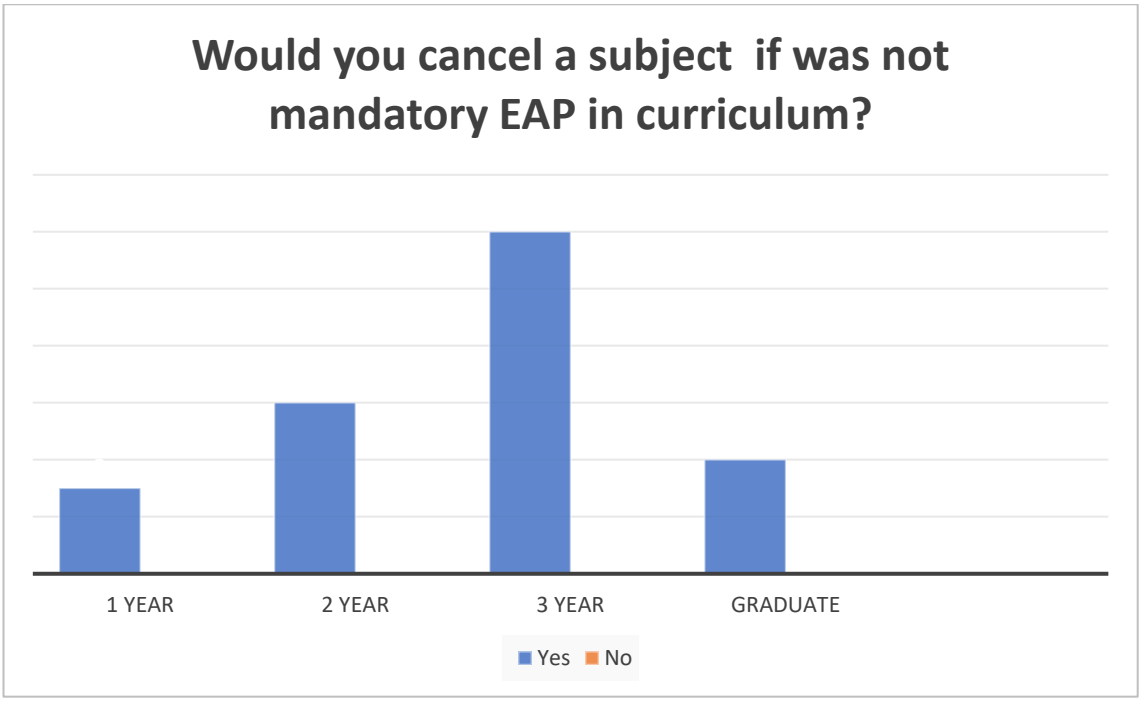

Figure 7. *Respondents' answers to the question.*

Further, the author asked questions about whether it is convenient to find courses and whether the description is clear.

It turned out that the 40 percent of respondents learned about the course they were interested in not from the system, but from friends or lecturers. Also, mainly IT students are dissatisfied with the description of subjects, almost all of them said that the description of a subject does not correspond to the reality.

Students and graduates were asked same questions, the interview consisted of 10 questions, to which students gave detailed answers.

It should be noted that humanitarian specialties have a practice base - that is, a student will find a practice in any case. Whereas in the specialty of Information Technology, the students are forced to look for practice themselves and this often creates difficulties.

In this study, the author paid attention to the issue of finding practice specifically for IT students, for the above reasons.

The next block of questions is related to practice, and it will consider in more detail how IT students find practice. 8 percent found an internship with the help of lecturers, 20 percent found an internship with the help of acquaintances, the rest used popular websites such as cv.ee, cvkeskus.ee, and the websites of companies.

The diagram below shows where IT students find practice.

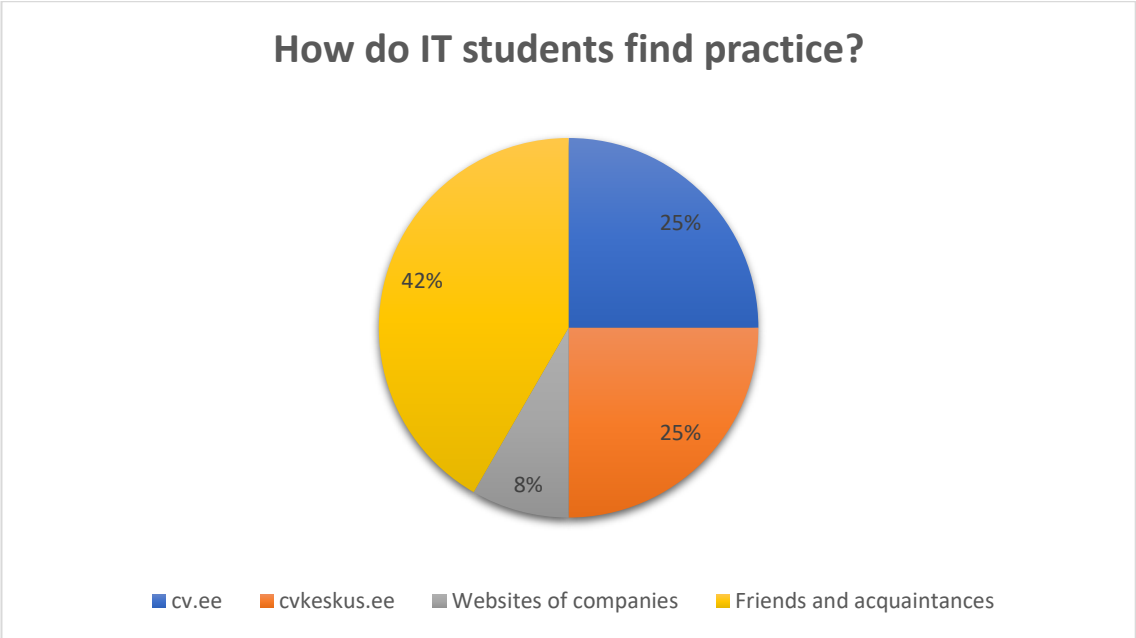

Figure 8. *Respondents' answers to the question.*

Such a search for practice is inconvenient, person needs to look through applications on many websites, find companies that are nearby, go to their websites and look through vacancies. 83 percent of respondents find this method inconvenient and would prefer to see vacancies in one place.

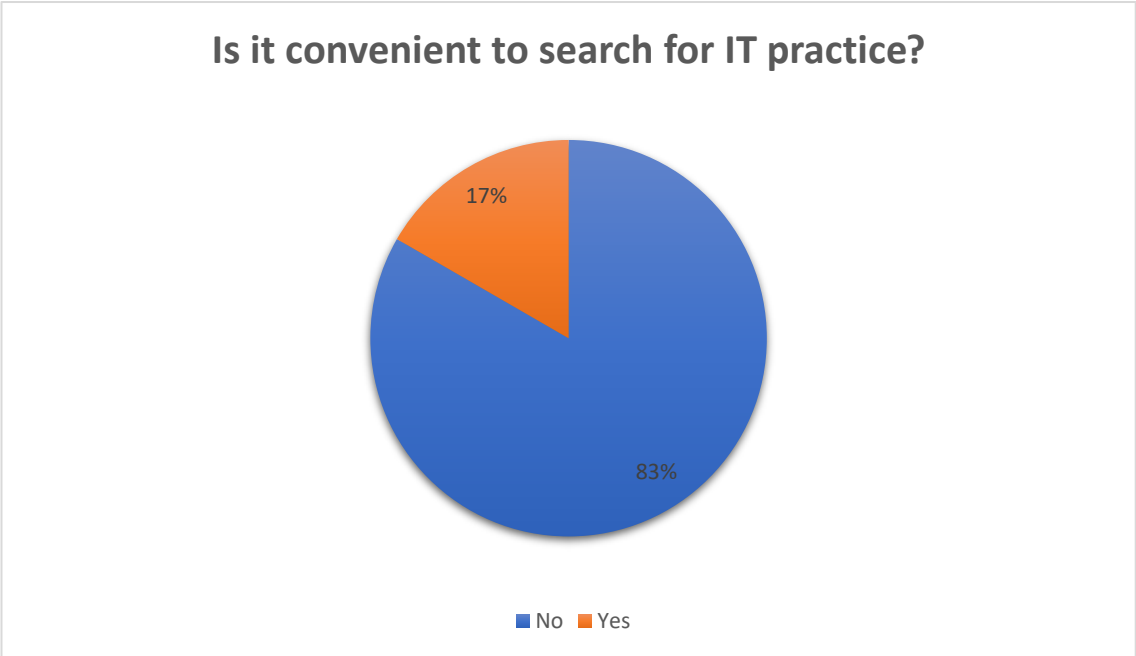

Figure 9. *Respondents' answers to the question.*

Furthermore, students often take courses they are interested in, such areas such as coursera.org, udemy.com and w3schools.com. As well as other sites or online courses that are of interest to the individual students or that will help them complete the subjects in the curriculum.

Due to the coronavirus, students can take courses in other branches of the university, which was previously problematic. For example, it is possible to take subjects from the University of Tartu or Pärnu College.

But only 4% took advantage of this opportunity. And the main reasons distinguished: fear that the subject will be too complicated, language problems, and too much workload at that moment.

Also, IT students do not trust the description of a subject given in the SIS, since there are often no coincidences at Narva College, and they simply do not want to take risks.

The chart below shows this data:

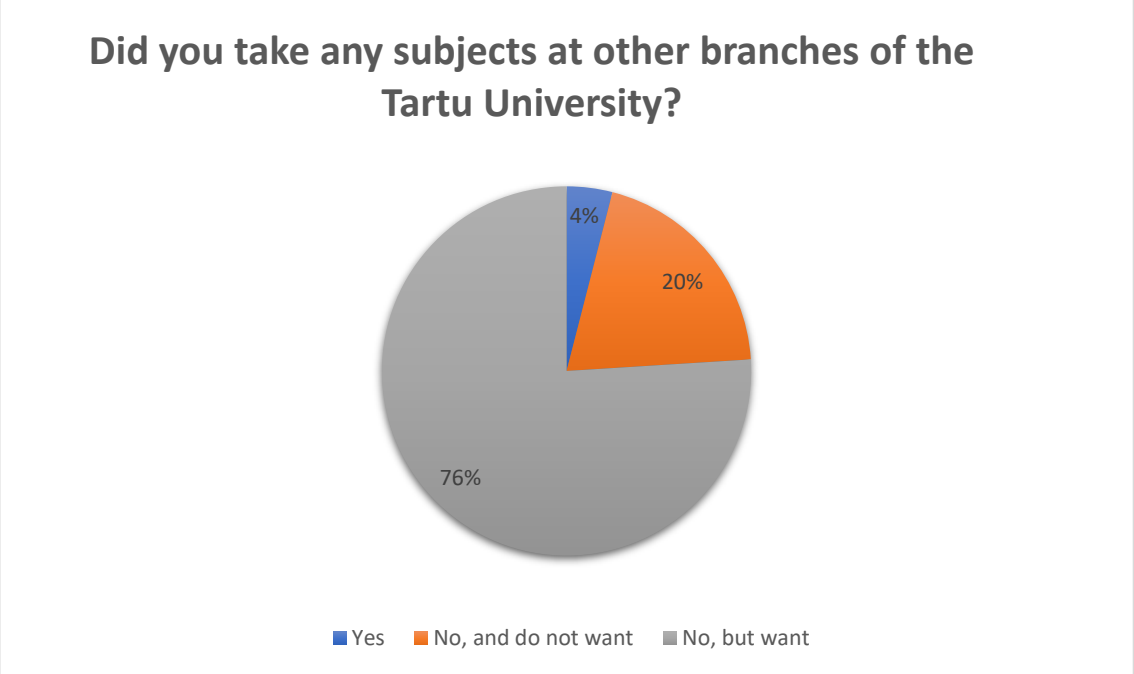

Figure 10. *Respondents' answers to the question.*

76 percent of the students would like to study subjects in other industries, but the reasons described above do not allow them to do so. Likewise, the technical students will study subjects, even if they exceed the workload, but will be interesting or useful for them.

Humanities students generally do not want to exceed their academic workload. Also, the author asked the teachers, and they answered that the students who have chosen a subject that does not suit them, bring small problems to the group. For example, this applies to group work or workshops.

After analyzing the data, the author of this work concluded that it is necessary to create a system that would help students choose the best courses for them. Also, students need a place where they can see practice proposals in one place.

It should be noted that even though the humanities students have their own practice base, almost all of the surveyed students showed interest in the fact that it would be possible to choose a practice on their own.

Teachers and youth workers would be interested in getting an internship that suits their wishes and interests.

The Solution

One solution to this problem would be to create a system that will help students choose the course that suits them best at that moment. In that system, the program itself will select courses, work, or practice for the user, based on his achievements, successes, and hobbies.

Since the college has a system of compulsory and non-compulsory subjects, this system will select courses only from the modules "Electives" and "Optional Courses".

Very often, students are faced with the problem of choosing a practice, this problem is especially relevant for the students of the "Development of Information Technology Systems" specialty. But also, according to the results of the survey, it would be interesting and useful for students of other specialties, such as youth work and business and digital solutions to try the options selected specifically for them.

Also, in the future, the system will be able to select suitable job vacancies for students.

The Aim

The aim of this thesis is to develop a system that will help students choose the courses, practice and, possibly, work that is in accordance with their skills. Students will not need to register for subjects and then cancel them because they were too easy or difficult.

Also, students will be able to see all the available practice for their level, and they do not have to look for practice on different websites or other places. Also, it will not be necessary to ask acquaintances or friends to help with practice, sometimes problems arise because of this.

The system will also be useful to employers who immediately see the level of the student who comes to them for practice.

In addition, the system will be useful for teachers who will no longer have students in a group who have mistakenly registered for the subject. Their groups will have students who are really interested in the subject. In the future, this will increase the quality of knowledge and the quality of work. This is especially good for subjects where there is group or teamwork, because if there is a person in a team who is disinterested in the subject or work, then the rest of participants have a heavy load, which ultimately has a bad effect on the level of mastering a subject.

Also, lecturers will be able to conduct a small test so that students can determine their level and decide whether to register for this subject or not.

This can be especially useful for language subjects, where there may be certain narrow topics that can be difficult for most students.

The Motivation

The motivation for this thesis is that the system is needed to help three groups of people at the same time: students, lecturers, and the people offering jobs.

Students will find it convenient to select courses and practices that match their skills and desires. Also, they will be able to see what skills they will have to improve on to take an interesting course if it is unavailable.

Also, the system will help solve the problem with practice, selecting suitable vacancies for students in one place.

In the future, the student will be able to understand what courses and skills they need to improve to get an interesting practice. Students no longer must take the course, and then cancel it, because it was too complicated, simple, or uninteresting.

It will be convenient for lecturers to indicate the level that is required for this course. And students for whom the subject is too difficult or not interesting will not be included in their group.

For employees, this system is convenient and useful in that they can post an advertisement for practice, indicating the desired level of a candidate, and the system will show the best candidates.

Tasks

The main task is to create a system that will recommend courses based on user profile or basic information.

It is also necessary to create an administrator profile. The administrator will be able to control the entire operation of the system. Also need to create a database in which to store all user data, and system data. The user's personal data must be protected.

During the development of the system, it is important to create a working algorithm that will be understandable and basic. The code should be easily readable, understandable, and include comments, so that another specialist can work and continue developing the project in the future.

Needful to use reliable development products so that there is no situation that the product is no longer supported.

It is also necessary that enough documentation was released for the product, which would be on the public domain. The design of system should be intuitive to any user, so that a person, even far from technical solutions, would understand without instructions what to do.

The colors for the system should be soft, the text should be readable, the font should be pleasant to read.

There should be no abrupt transitions, the system should not overload the user with unnecessary information and unnecessary design.

IMPLEMENTED TECHNOLOGIES AND AVAILABLE SOLUTIONS

Technologies

1. ASP.NET -NET is a developer platform made up of tools, programming languages, and libraries for building many different types of applications. The base platform provides components that apply to all different types of apps. Additional frameworks, such as ASP.NET, extend .NET with components for building specific types of apps. (What is .NET?, 2021)
2. C# is a modern, object-oriented, and type-safe programming language. C# enables developers to build many types of secure and robust applications that run in the .NET ecosystem. (A tour of the C# language, 2021)
3. MS SQL Server Express is a powerful and reliable free data management system that delivers a rich and reliable data store for lightweight Web Sites and desktop applications. (Microsoft® SQL Server® 2019 Express, 2020)
4. NuGet - for .NET (including .NET Core), the Microsoft-supported mechanism for sharing code is NuGet, which defines how packages for .NET are created, hosted, and consumed, and provides the tools for each of those roles. (An introduction to NuGet, 2019)

Design and Architecture of Application

Application architecture is the division of the application into layers. Domain Model, that is, those model classes that are used in the application and whose objects are stored in the database, these are CourseId, CourseLevelId.

It is the outer layer and keeps peripheral concerns like UI. For a Web application, it represents the Web API. This layer has an implementation of the dependency injection principle so that the application builds a loosely coupled structure and can communicate to the internal layer via interfaces.

This project uses ASP.NET Identity. This is intended to replace the previous ASP.NET membership and simple membership systems. It includes profile support, OAuth integration, works with OWIN, and is part of the ASP.NET templates that ship with Visual Studio 2013. (Introduction to ASP.NET Identity, 2019) ASP.NET Identity is distributed as a NuGet package installed in ASP.NET MVC, Web Forms, and Web API templates that ship with Visual Studio 2017. This NuGet package can be downloaded from the NuGet Gallery.

There are Models, Views, and Controllers, as well as additional folders for Data and Services.

Model objects are parts of the application which implement the logic for the application's data domain. It retrieves and stores model state in a database.

Views are the components which are used to display the application's user interface also called view model in MVC. It displays the .Net MVC application's which is created from the model data.

Controllers handle user interaction, work with the model, and select a view to render that display Ul. In a .Net MVC app, the view only displays information, the controller manages and responds to user input & interaction using action filters in MVC. (ASP.NET MVC Tutorial for Beginners, 2021)

In the figure below can see the architecture of the project.

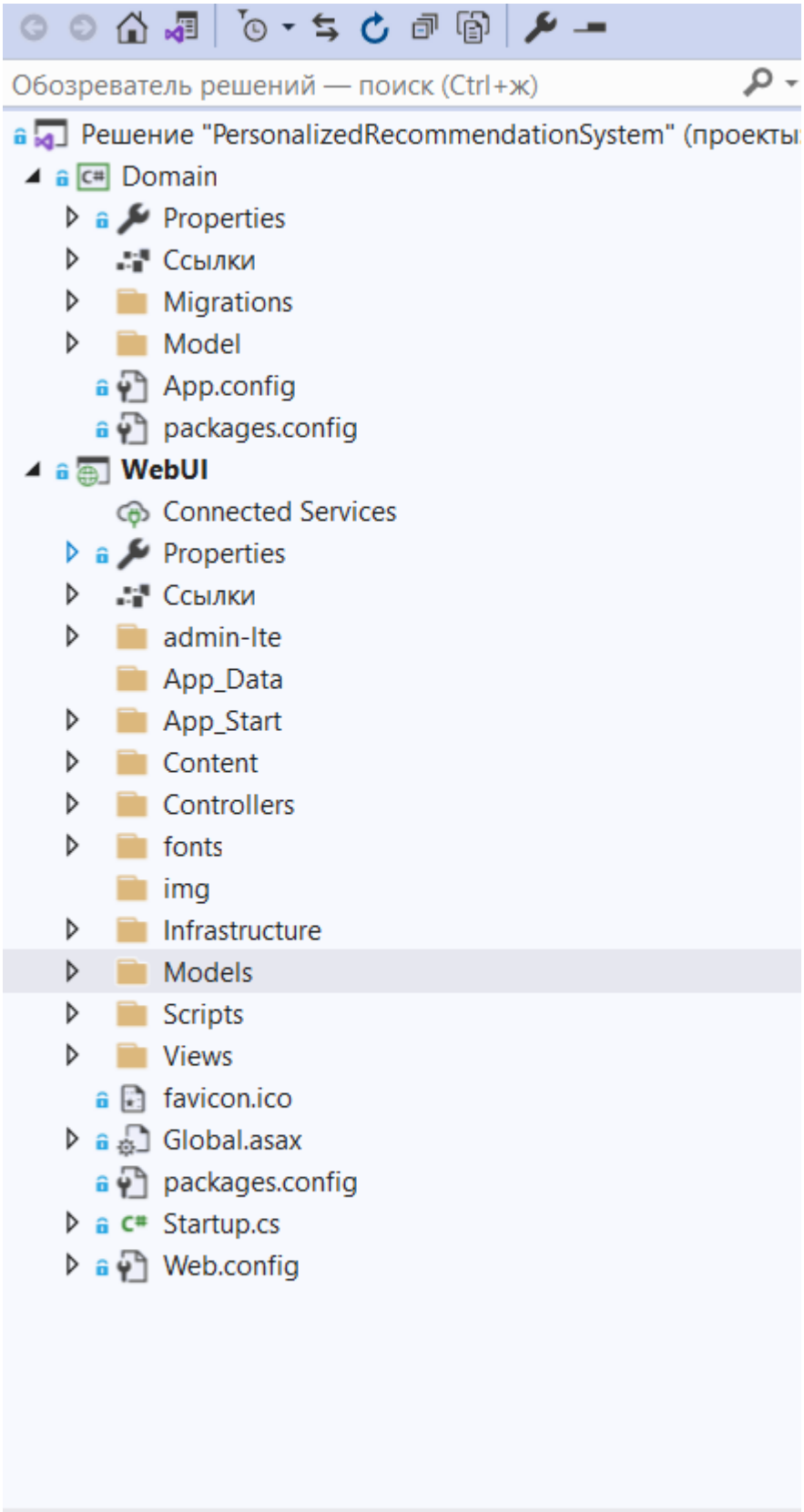

Figure 11. *Project architecture.*

The project uses ASP.NET Identity. ASP.NET Identity is ASP.NET's built-in authentication and authorization system. This system allows users to create accounts, authenticate, manage accounts.

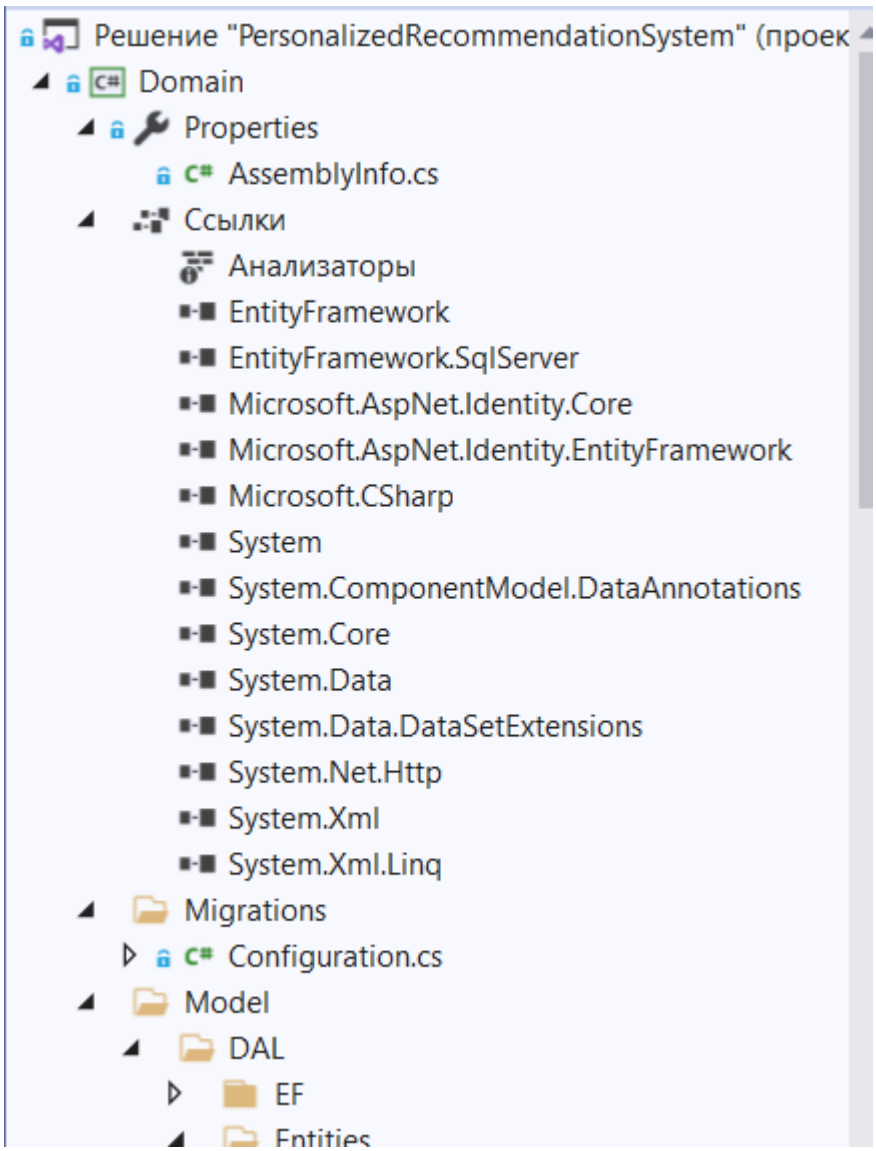

Figure 12. *Project architecture.*

This is a series of ASP.NET Identity Libraries:

- **Microsoft.AspNet.Identity.EntityFramework:** Contains Entity Framework classes that use ASP.NET Identity and communicate with the SQL Server.
- **Microsoft.AspNet.Identity.Core:** Contains a number of key ASP.NET Identity interfaces. The implementation of these interfaces will allow go beyond MS SQL Server and use other DBMSs, including systems, as account storage NoSQL;

Requirements

The system will have 4 types of users: students, lecturers, employees of firms, and administrator.

Students will be able to create their own profile in which they can indicate their skills and preferences. The system will present students with subjects that will be selected based on their data. For example, if a student indicates that they are interested in Russian but do not indicate their knowledge of Russian in their skills, the system will select for them the simplest subject with the lowest level of Russian so that the student can start learning.

Also, if a student indicates their knowledge of the Java programming language at an intermediate level in their skills, the system will select an intermediate or higher-level subject for them so that the student can develop and consolidate the skills they already have.

The system will also show suggestions for practice that are already suitable for the student based on his data. The student will be able to apply or contact the firm's representatives on their own.

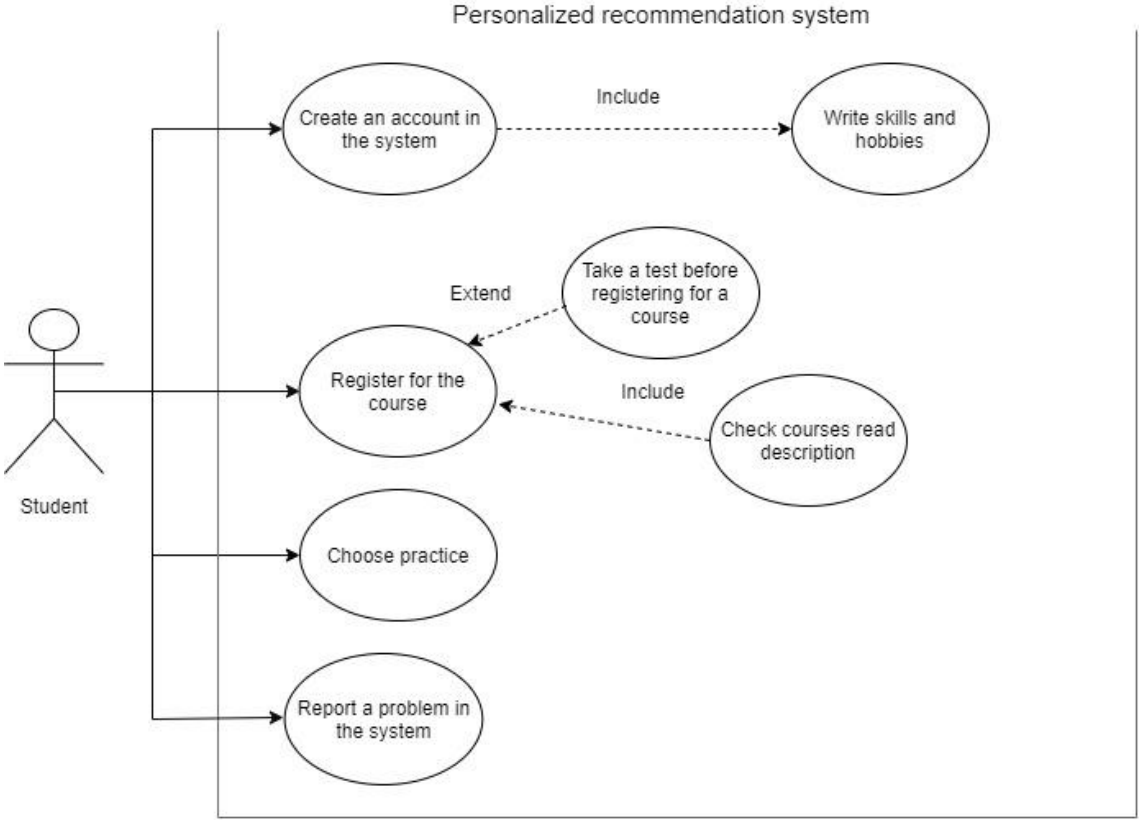

Figure 13. *Use-case diagram for student.*

This will prevent the student from confronting issues of choosing what they need to study, the many portals to find work and practice, and will save time. The student can change their preferences at any time.

Also, the student will need to pass tests to confirm their level of knowledge before some courses, where a lecturer considers such a test necessary.

The student use case diagram up shows how a student can use this system.

An administrator will create profiles for lecturers. This will reduce the occasional obstacles from lecturers with no technical skills. It should help lecturers who need help.

As we can see from the diagram below, lecturers need to go to the profile and create a course. If necessary, lecturers can create a test to test the knowledge of students, so that the students do not choose the subject by mistake and know exactly what will be in that course.

At the same time, lecturers need to choose the level of students and write a description of the course, including a description of the modules and homework that will be in this course.

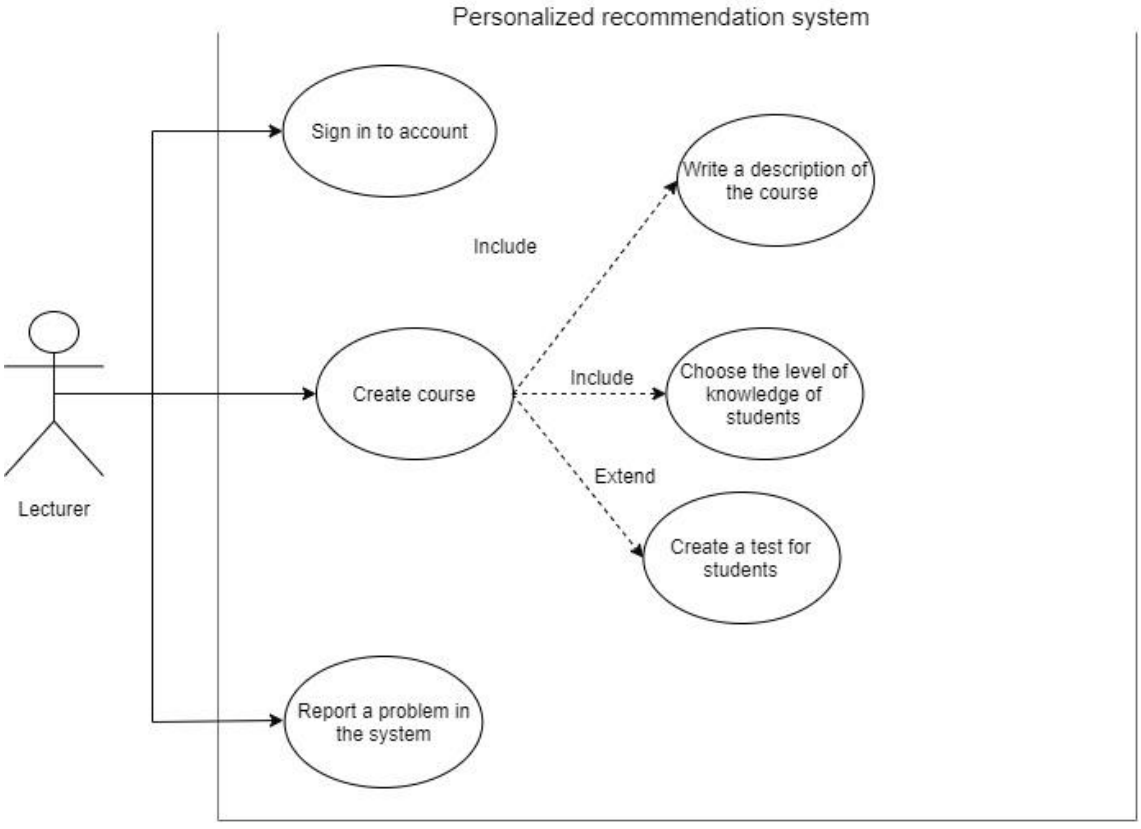

Figure 14. *Use-case diagram for lecturer.*

After completing a profile, employees of firms will be able to create offers for internship, in which they can indicate the necessary skills for students. The system will select the

best candidates for them, and employees will be able to contact the candidates they are interested in, just as the students will be able to contact the employees.

Employees, like teachers, will be able to create tests with which they can assess the level of knowledge of students. This will undoubtedly help them find the best candidates.

The diagram below shows how employees of companies can contact the system.

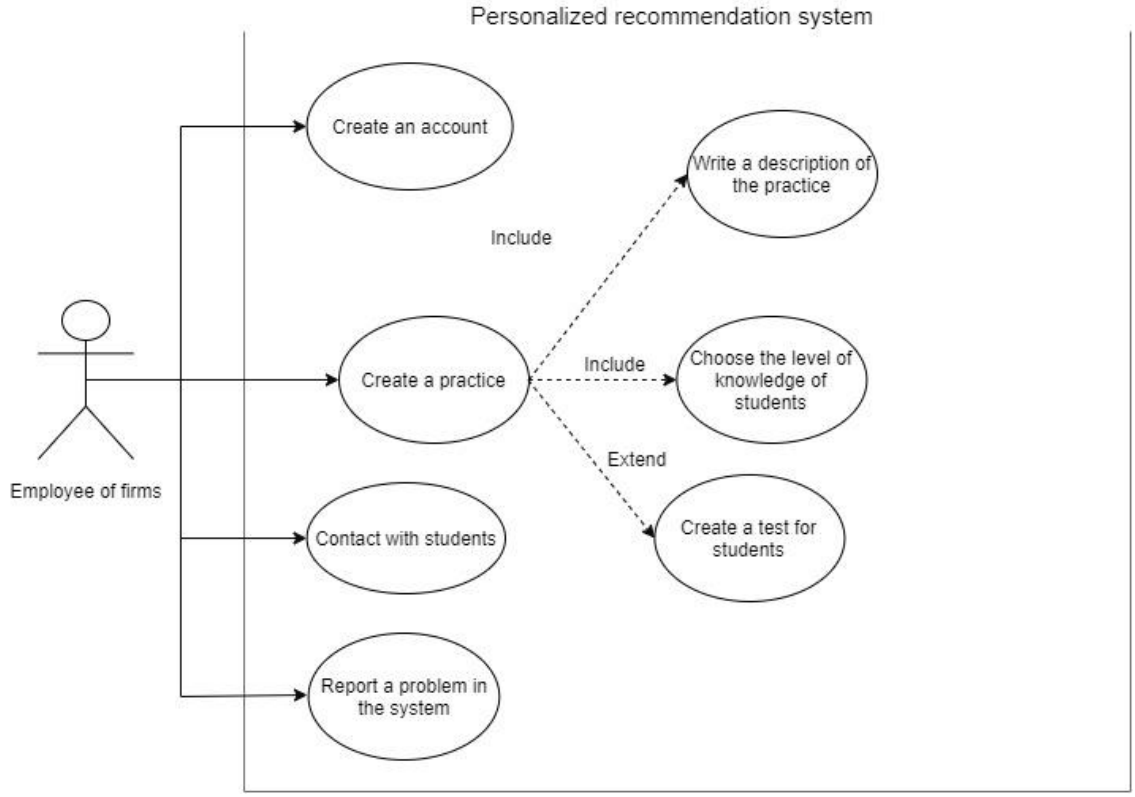

Figure 15. *Use-case diagram for employee of firms.*

The administrator must set up an account for lecturers, and to give them the information to access their account. Also, the administrator can restore access to user accounts if they have problems logging in. The administrator can also add / remove courses, change data and see all user data.

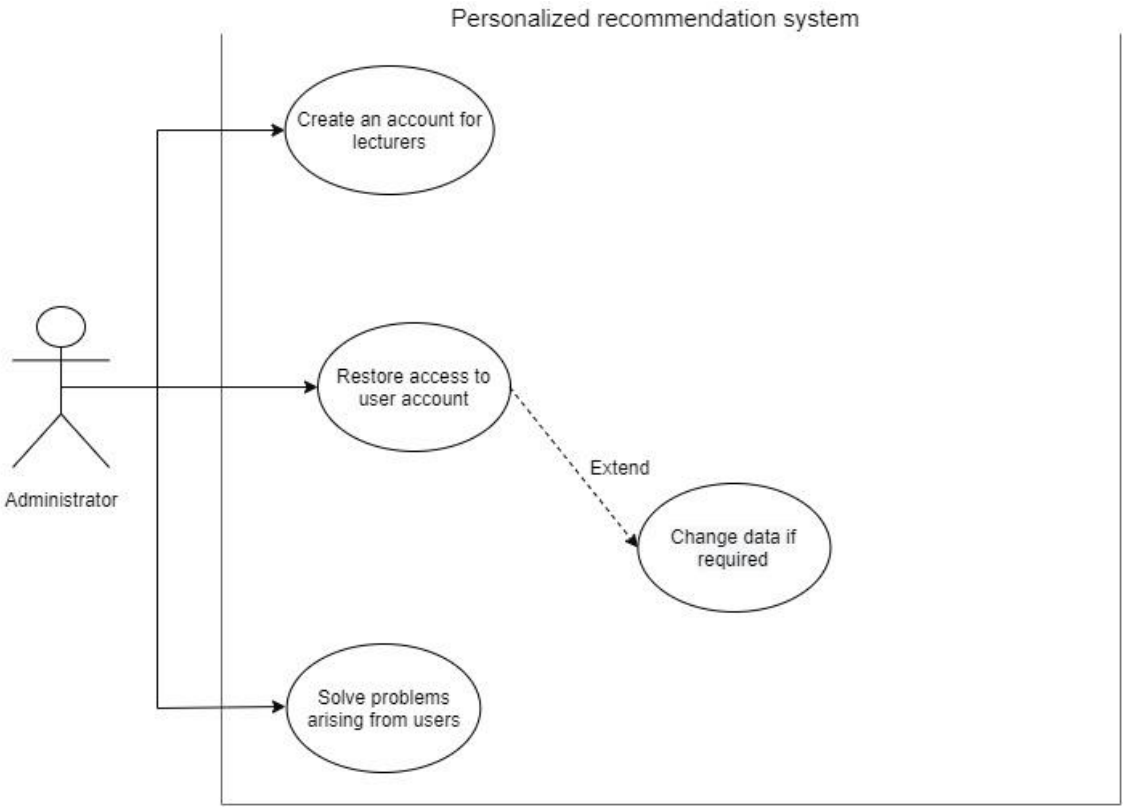

Figure 16. *Use-case diagram administrator.*

Functional Requirements

- The system is supposed to log in and log out;
- The system must be able to regain access if the user has forgotten the data;
- The system must include an administrator role;
- The system must include the user's role;
- The system must include at least three levels of courses (low, middle, and high);
- The system should provide the ability to create and edit courses.

Non-Functional Requirements

- The application must maintain high responsiveness with simultaneous access by several users;
- The system must have a clear and friendly interface;
- The maximum acceptable response time for web approves is 5 seconds;
- A single database server must quickly service queries;
- Only the administrator can see users' data;
- The new user account should be created in less than 10 seconds.

Minimum Viable Product

To solve the research problem within the framework of this thesis, a minimum viable product was developed. It includes the functions of choosing courses for users, the ability to register users, as well as the work of an administrator.

The administrator can add course levels, courses, new users, and delete users. Also, the administrator has user data, their email address, name, and phone number.

The user can create an account for themselves with a choice of their knowledge level for example low, medium, or high. The user can indicate their skills and interests, and the system will select suitable courses for them.

The screenshot below shows the home page of the application.

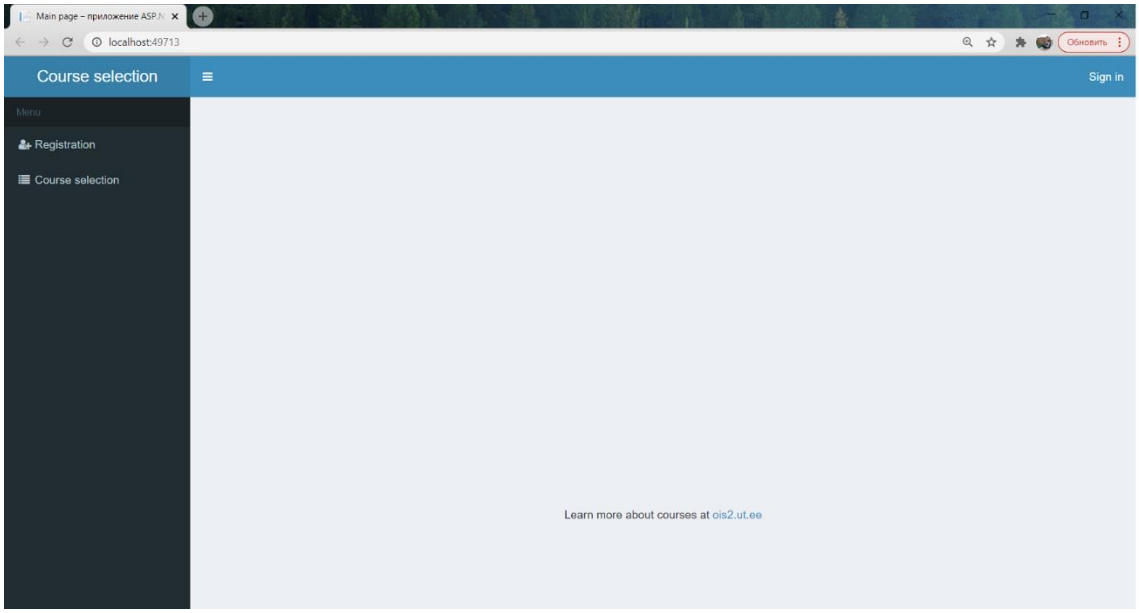

Figure 17. *Application home page.*

Figure 18 shows the user data available to the administrator. The administrator can see the name, e-mail address, phone number, user level. The administrator can also delete users.

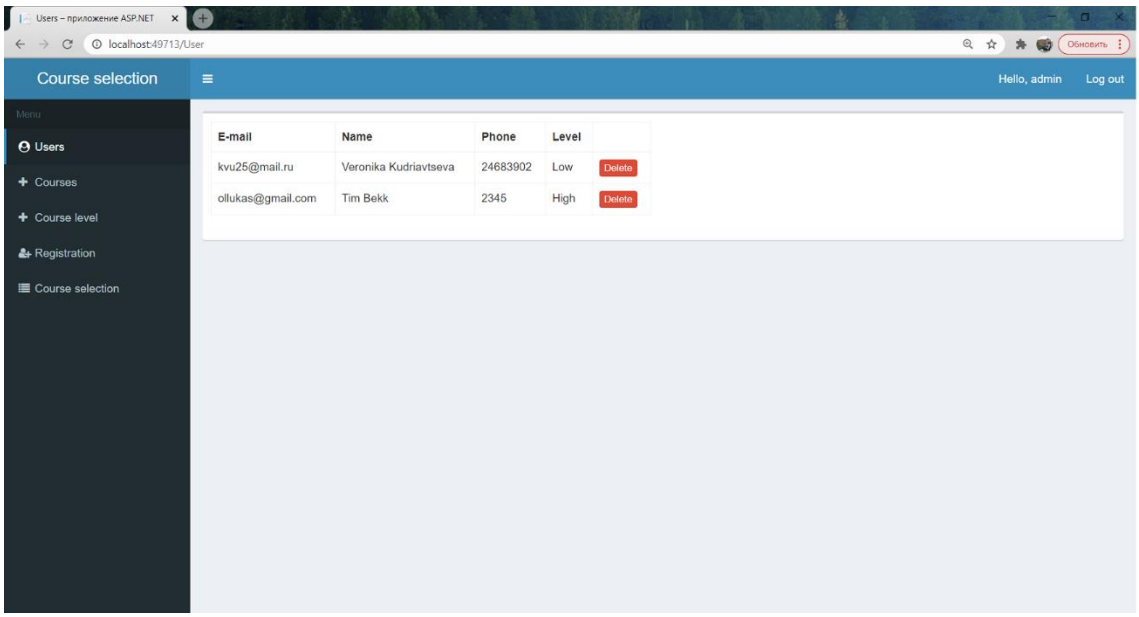

Figure 18. *User data page available to the administrator.*

Figures 19 and 20 show suitable courses for students at the moment.

As can see, the application notifies the user if there are no courses available currently.

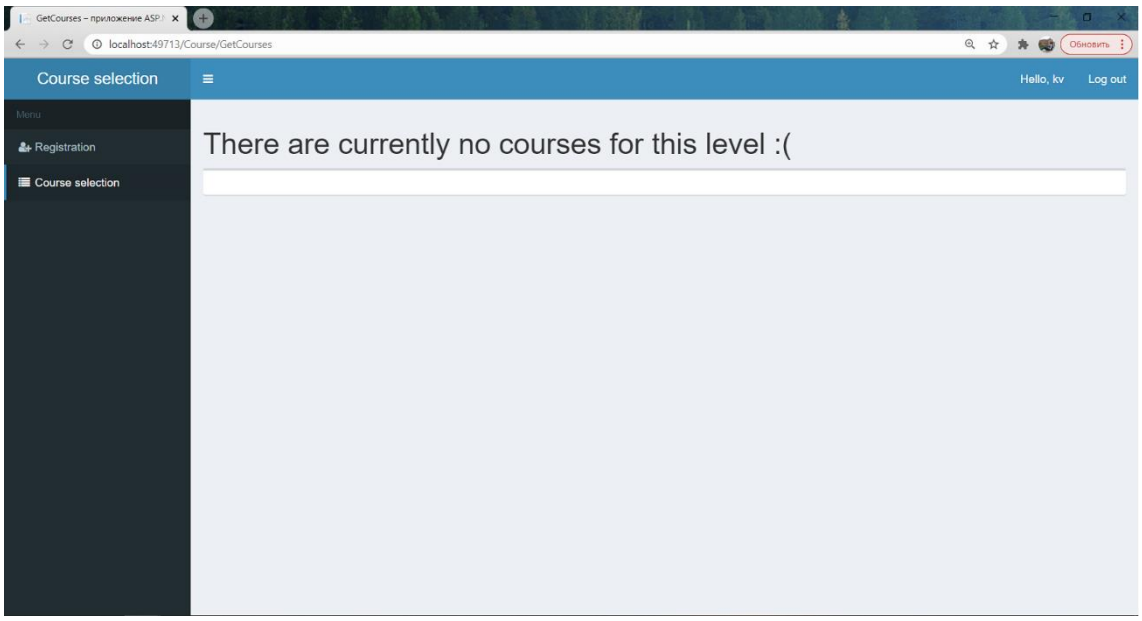

Figure 19. *Page with available courses for the user.*

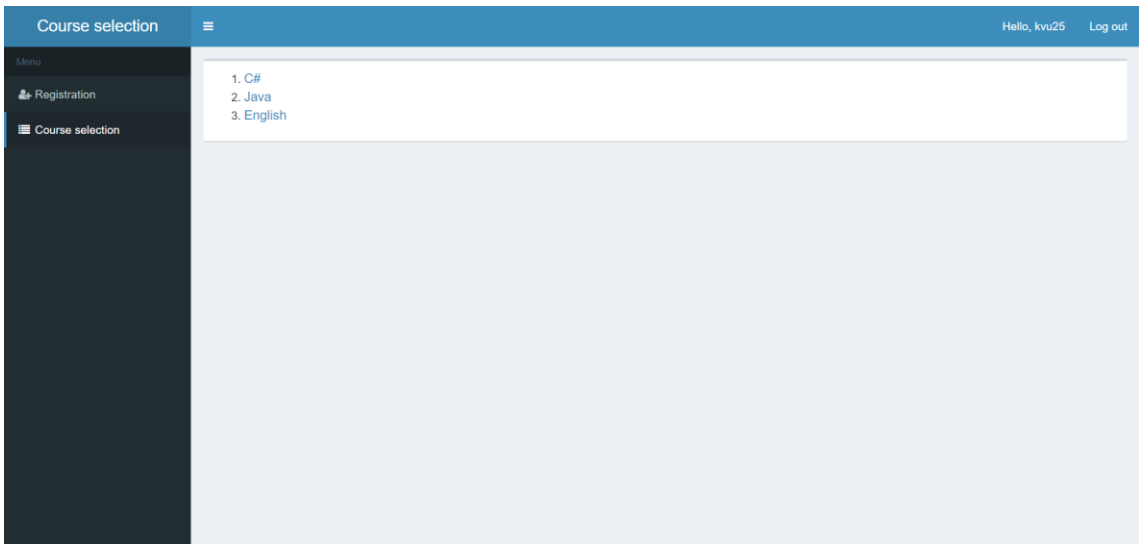

Figure 20. *Page with available courses for the user*.

Figure 21 presents a questionnaire, or information about a user. Here the user can see their answers and change their details.

On this page the list view of abilities and interests of users. Also contact information.

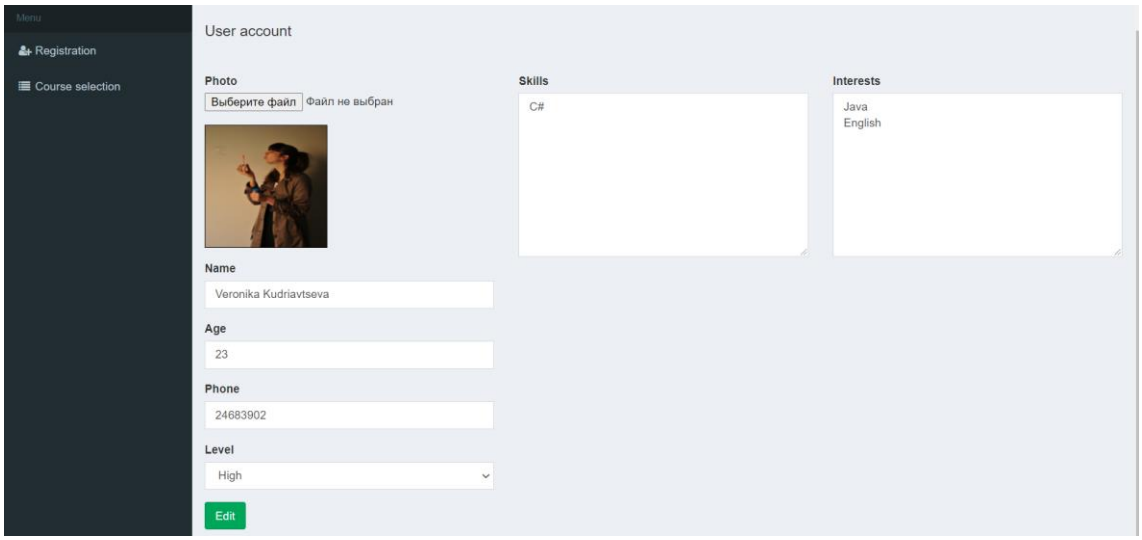

Figure 21. *User account page.*

Figure 22 shows a course management page. Here you can view the entire list of existing courses, edit, or delete courses.

Also, on this page, the administrator can add courses.

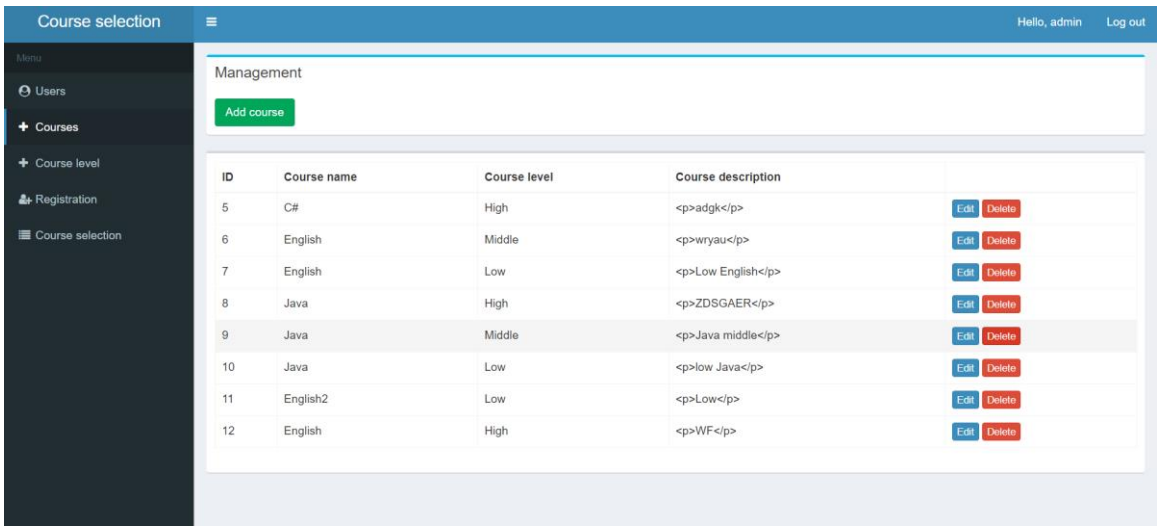

Figure 22. *Management page.*

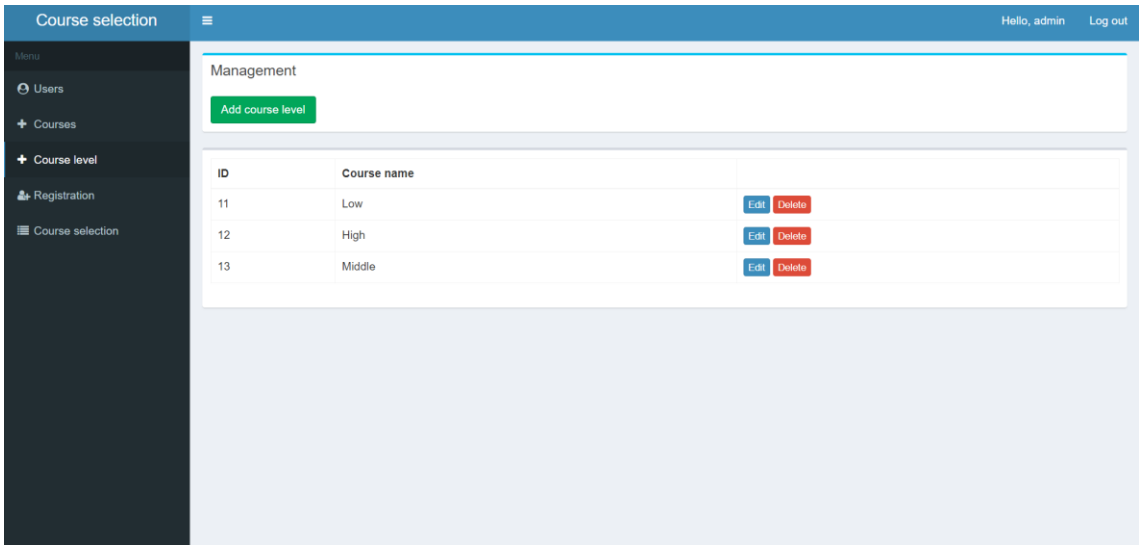

Figure 23. *Management page.*

Figure 22 also shows the management page. On it can see the user levels. Currently there are three of them, but the administrator can add more. For example, it is possible to add the levels of language proficiency, such as low, middle, and high. The administrator can also edit or delete levels.

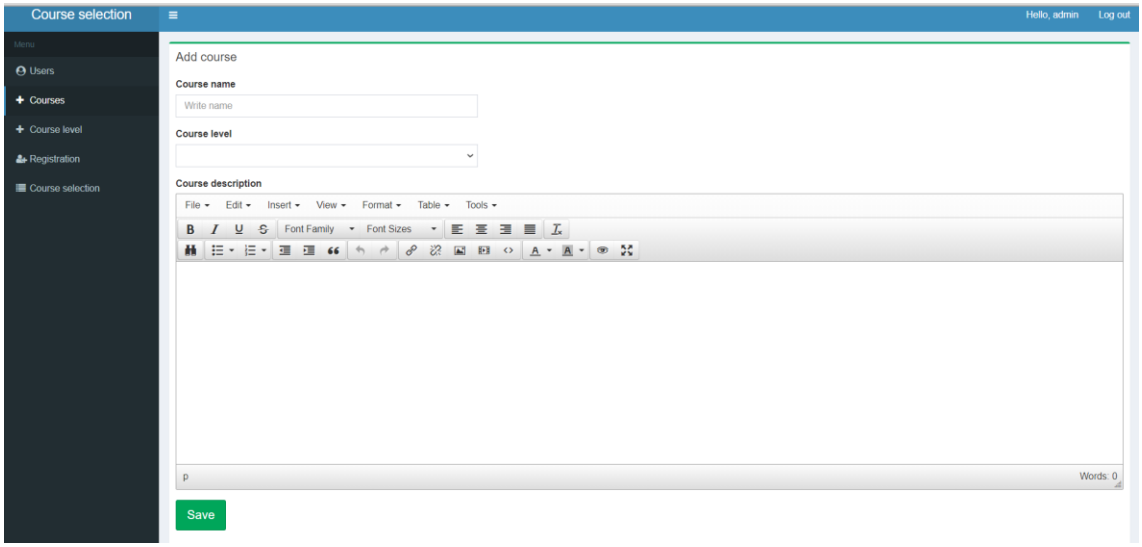

Figure 24. *Add course page.*

Figure 23 shows the add course page. Using it, the administrator can add a course. To do this, they need to write a name of the course, select the level of the course: low, medium, or high. And write a description of the course.

Application Classes

This thesis presents a minimum viable product that allows two types of users to work - student and administrator.

Below is the pseudocode for the student course matching function.

```
public function getCourses(){
    user <- get user information from database;
    courses <- get information about all courses from database;
    output <- create empty array;
    if user role == is user{
        foreach course in courses {
            if user.userLvl = course.courseLvl{
                output.add(course)
            }
            else {
                continue;
            }
        }
    }
}
```

Database Architecture

The sequence diagram (Figure 24) shows how data is transferred between the user, the application, and the database.

In the diagram, the user sends a request, after the app sends a request to the Controller. When a request is received, it selects the correct controller to process the request and passes the request data to it. The controller processes this data and sends back the processing result. Controller sends a request to Model. Models are one of the key components of Model-View-Controller. The key task of the models is to describe the structure and logic of the data used.

As a rule, all used entities in the application are separated into separate models, which describe the structure of each entity. Depending on the tasks and subject area, it is possible to distinguish different number of models in the application. Model sends a request to the database.

The database sends the response to model. The Model sends a response to the Controller, and the Controller sends it to the View, which sends the response to the user in the form of an http page.

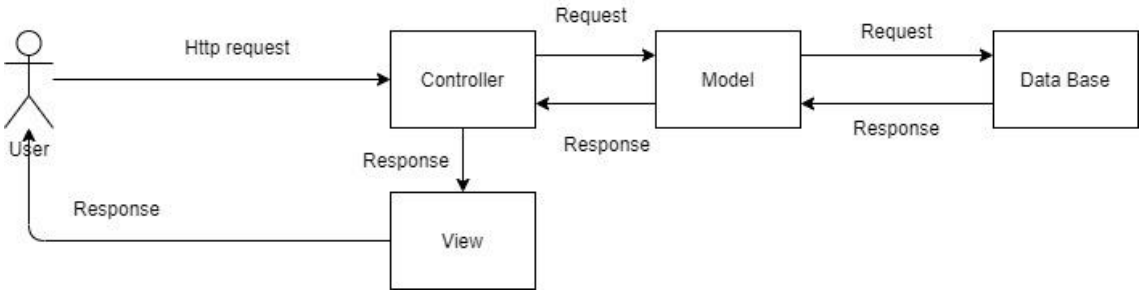

Figure 25. *Sequence diagram.*

Figure 25 is a show the architecture of the database.

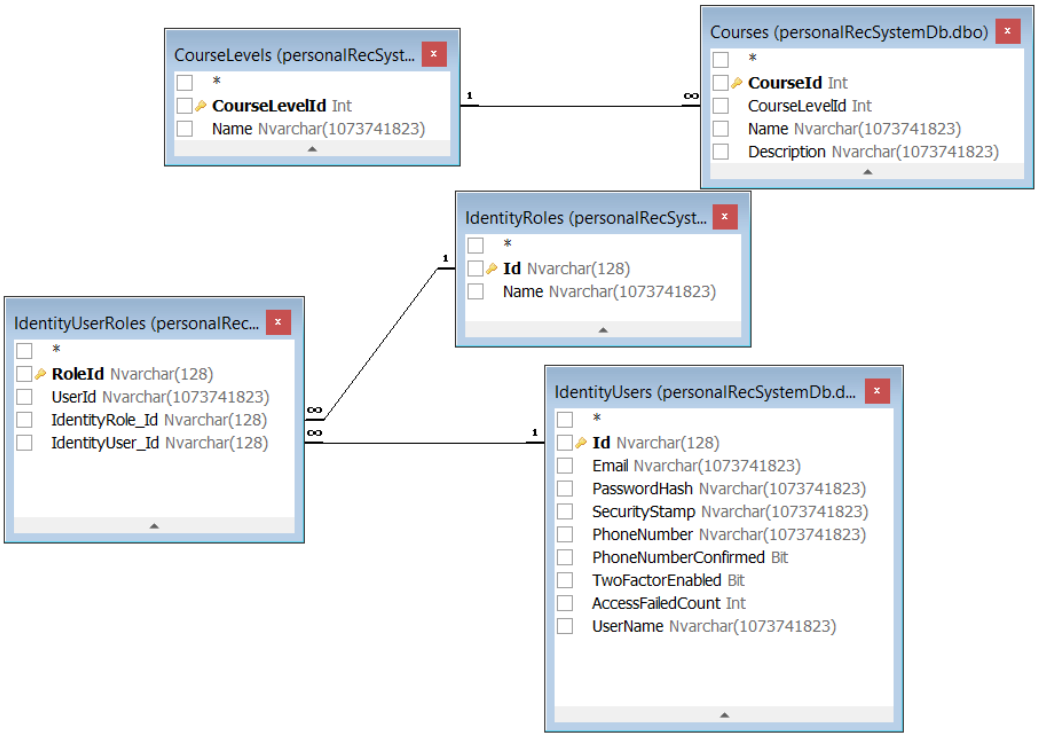

Figure 26. *Database architecture.*

The project database contains 5 tables. Their properties are described below.

«Courses» is a source table created to store a courses data. This table contains the following fields:

- CourseId - integer, unique index or primary key.
- CourseLevelId - integer, unique index or primary key.
- Name - Name of course, Nvarchar, variable-size string data, maximum storage size is (up to 2 GB).
- Description - Course Description, Nvarchar, variable-size string data, maximum storage size is (up to 2 GB).

Table «CourseLevels» designed to store course levels. This table contains the following fields:

- CourseLevelId- integer.
- Name- Name of course, Nvarchar, variable-size string data, maximum storage size is (up to 2 GB).

«IdentityRoles» this is a table that stores identity user roles and table contains the following fields:

- Id- Nvarchar.

- Name – User role name, Nvarchar.

«IdentityUserRoles» contains data about the personal role of users. This table contains the following fields:

- RoleId - the user role id is stored here, Nvarchar.
- UserId - the user id is stored here, Nvarchar.
- IdentityRole_Id - here the identity role id of the user's role is stored, Nvarchar.
- IdentityUser_Id - here the identity user id of the user's role is stored, Nvarchar.

The next table «IdentityUsers» stores personal data of users. This table contains the following fields:

- Id - the user id is stored here, Nvarchar.
- Email - the user email is stored here, Nvarchar.
- EmailConfirmed - the user confirmed email is stored here, bit.
- PasswordHash - The user's password is stored here, when a password has been “hashed” it means it has been turned into a scrambled representation of itself, Nvarchar.
- SecurityStamp - the security stamp is stored here, Nvarchar.
- PhoneNumber - It contains the user's phone number, Nvarchar.
- PhoneNumberConfirmed - It contains the confirmed user's phone number, bit.
- TwoFactorEnabled - This is where the data for two-factor enabled is stored, bit.
- AccessFailedCount - Here are stored the data of the failed user logon attempts, int.
- UserName – User Name, Nvarchar, variable-size string data, maximum storage size is (up to 2 GB).

Future Plans

Notice: The author will continue to work on the system, conduct testing with students, create profiles of lecturers and employees, and develop the system.

This paper presents the MVP. In the future, it will be possible to work with all four types of users: students, lectures, employee of the companies and administrator.

Also, the system can be developed not only for the Narva College of the University of Tartu, but also for nationwide usage. In this case, students will have more opportunities to find suitable practice, and they will also be able to choose a subject in another building or even from another university.

The system will preserve the student's progress, protect all work, participation in various events and courses from other educational sites.

All these features will help students choose the ideal study plan, as well as help them find interesting and suitable practice without stress and with little to no mistakes. And also, the system will recommend appropriate subjects. Compared to currently used SIS by University of Tartu where students can not add any background information about themselves, nor does it recommend suitable subjects, the final product will ask about students' skills and interests and will recommend appropriate subjects by using a keyword system.

In terms of finding practice, the final product will recommend practice that matches students' skillset. Currently used SIS by University of Tartu does not offer this opportunity.

Also, this system can be integrated with the existing information system of the University of Tartu. It will not be a separate system from UT SIS, it will expand functionality of the already working UT SIS.

At the moment, the registration for subjects, especially for elective subjects, and for additional subjects, is that the student must write the subject code in the registration form and select it. By adding the functions to this system, the SIS will become more convenient and useful.

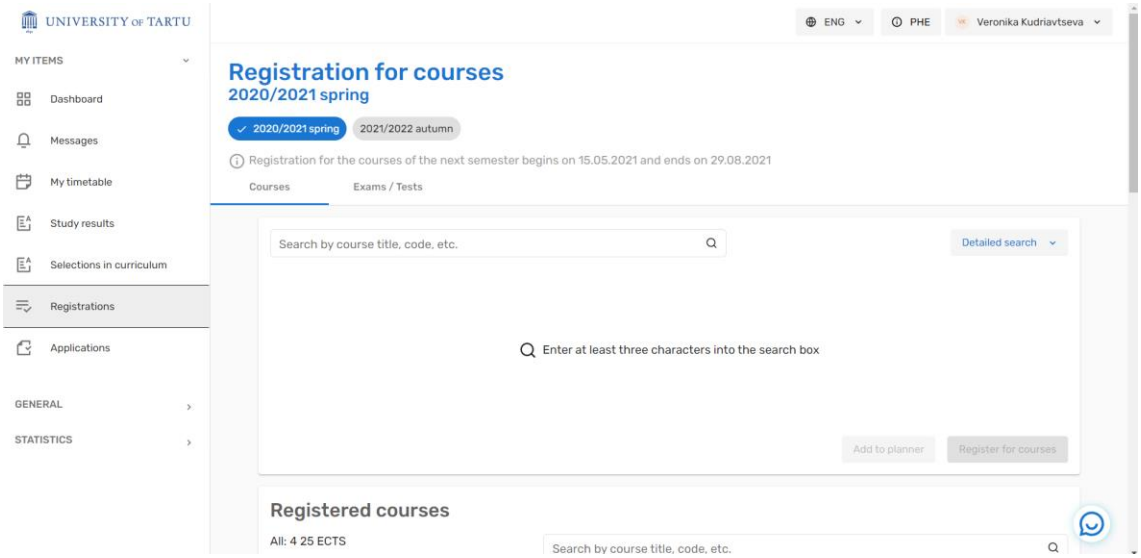

Figure 27. *TU SIS registration page.*

Also, if the two systems are connected, then can add access to the database and lecturers will not have to create subjects, but they will be able to create tests to test the knowledge of students before they register to this subject.

SUMMARY IN ESTONIAN

Lõputöö teema on «PERSONALISEERITUD SOOVITSSÜSTEEM KÕRGKOOLI ÕPPEJÕUDUDE ABISTAMISEKS».

Lõputöö probleemina püstitati, et töö kirjutamise hetkel oli õppeinfosüsteemis probleem sobivate õppeainete valimisega. Probleemile lahenduse pakkumiseks viidi läbi täpsustav uuring erinevate Tartu Ülikooli Narva Kolledži üliõpilaste seas, et välja selgitada õppeainetele registreerimise tagamaid, eestkätt põhjuseid, miks on õppeained valitud ja/või registreerimine tühistatud.

Erinevate tudengite poolt antud vastustest selgus, et kõigil küsitletud üliõpilastel on raskusi ainete valimisega valikaine ja vabaaine moodulitest. Samuti on eriala "Infotehnoloogiliste süsteemide arendus" üliõpilastel raskusi praktika leidmise ja valimisega.

Küsitluse tulemustele tuginedes püstitas töö autor eesmärgi arendada esialgne variant süsteemist, mis võimaldab üliõpilastel leida kursusi, praktikat ning tööd, mis on vastavuses nende võimete ja huvidega. Lõputöö praktilise töö käigus valmis süsteem, millega on võimalik luua üliõpilase profiil, sisestada oskused ja huvid ning süsteem valib sobivad kursused. Samuti saab süsteemis töötada administraator, kes saab luua ja kustutada kursusi, profiile, samuti redigeerida kursusi ja vaadata kasutajaandmeid.

Lõputöö jooksul valminud süsteemi on võimalik tulevikus edasi arendada, et lisada funktsioone ning lektorite profiilid, et nad saaksid luua kursusi üliõpilastele. Süsteemi saavad kasutada ka ettevõtete töötajad, kes saavad üliõpilastele praktikapakkumisi luua.

Lisaks saab integreerida arendatavat süsteemi Tartu Ülikooli ÕISiga nii, et laiendatakse olemasoleva õppeinfosüsteemi funktsionaalsust.

See lõputoitöö koosneb sissejuhatusest ja peat ükist. Sissejuhatus tutvustab probleemi ja analüüsib saadud andmeid uurimisrühma. Peatükk keskendub süsteemile, selle struktuurile, samuti valitud tehnoloogiatele. Süsteemi tulevase arengu võimalused on seal välja toodud.

REFERENCES

1. *A tour of the C# language*. (2021, 01 28). Retrieved 01 30, 2021, from Microsoft: https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/
2. Addie, S. S. (2019, 12 05). *Handle requests with controllers in ASP.NET Core MVC*. Retrieved 03 12, 2021, from docs.microsoft.com: https:///docs.microsoft.comru-ru/aspnet/core/mvc/controllers/actions?view=aspnetcore-5.0
3. *An introduction to NuGet*. (2019, 05 24). Retrieved 01 12, 2021, from docs.microsoft.com: https://docs.microsoft.com/en-us/nuget/what-is-nuget
4. Microsoft. (2019, 07 19). *nchar and nvarchar (Transact-SQL)*. Retrieved 01 25, 2020, from Microsoft: https://docs.microsoft.com/en-us/sql/t-sql/data-types/nchar-and-nvarchar-transact-sql?view=sql-server-ver15
5. Microsoft. (2019, 03 19). *Welcome to the Visual Studio IDE*. Retrieved 10 28, 2020, from docs.microsoft.com: https://docs.microsoft.com/ru-ru/visualstudio/get-started/visual-studio-ide?view=vs-2019
6. *Microsoft® SQL Server® 2019 Express*. (2020, 3 24). Retrieved 10 27, 2020, from Microsoft: https://www.microsoft.com/en-us/download/details.aspx?id=101064
7. Professor Web. (2021). *https://professorweb.ru/*. Retrieved 04 01, 2021, from Уроки по C# и платформе .NET Framework: https://professorweb.ru/
8. Shetty, B. (2019, 07 19). *AN IN-DEPTH GUIDE TO HOW RECOMMENDER SYSTEMS WORK*. Retrieved 01 20, 2021, from builtin: https://builtin.com/data-science/recommender-systems
9. *What is .NET?* (2021). Retrieved 01 15, 2021, from Microsoft: https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet
10. *Синтаксис HTML.* (2016, 06 23). Retrieved from zametkinapolyah: https://zametkinapolyah.ru/verstka-sajtov/sintaksis-html-html-tegi-parnye-html-tegi-odinochnye-html-tegi-chto-takoe-html-teg.html#HTML__HTML

Appendix 1

The project files are located here:

https://drive.google.com/drive/folders/1iZqU52_gFtFFbwuDL5GLQ_Nrdcsc5oMy?usp=sharing